HILL HIGH SCHOOL DATABASE MANAGEMENT ANALYSIS (GRADE 12)
(SQL ANALYSIS)

Data Analyst: Odudu Ochuko Richie

INTRODUCTION

A student database is important for ensuring efficient administration, tracking student progress, facilitating communication, and making informed decisions regarding academic planning and support services.

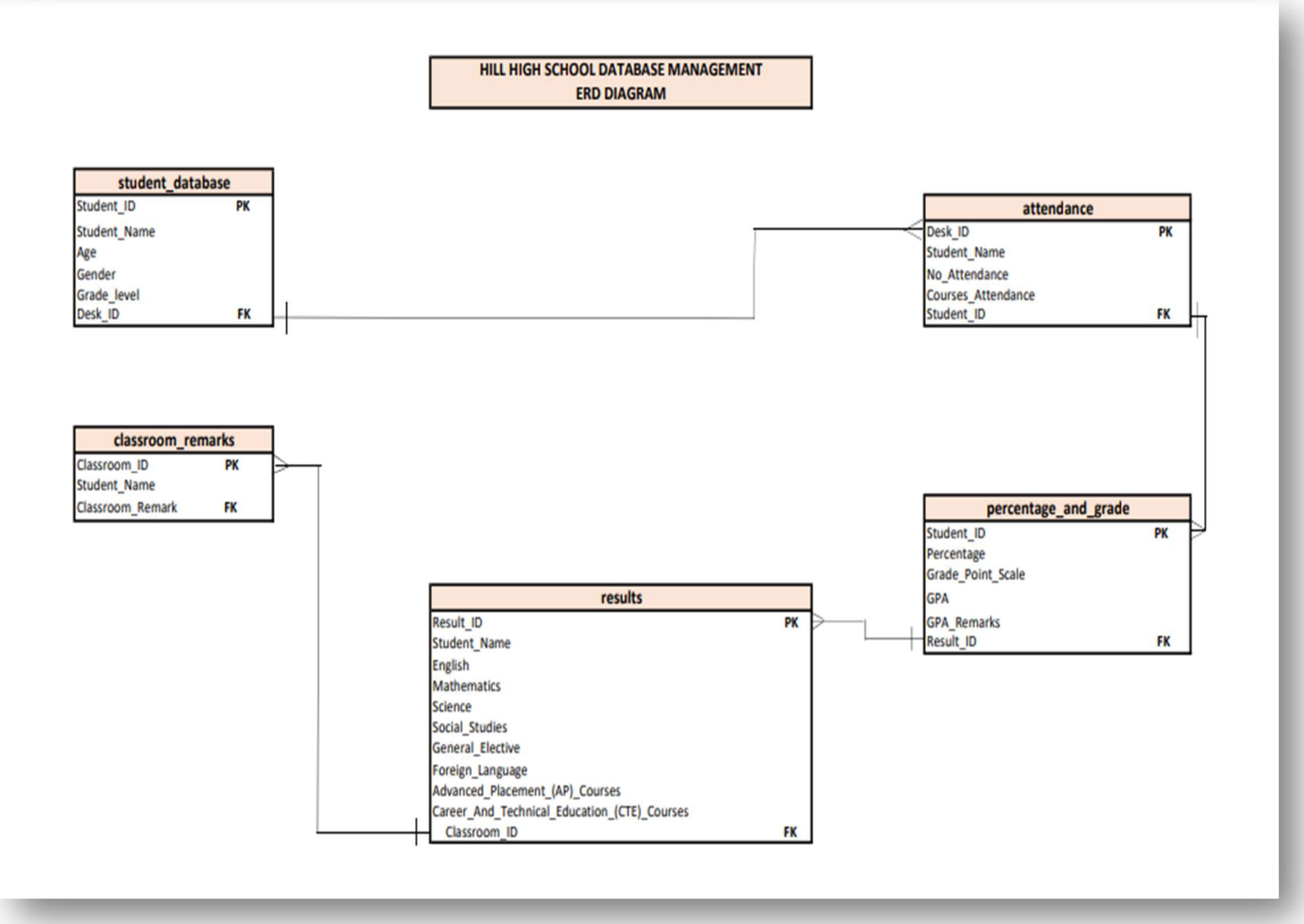
This is an SQL analysis on a student database with a primary focus on grade 12 students. This student database (for Grade 12) serves as a central repository for storing and managing various types of data related to Hill High School (Grade 12 students), including personal information, academic records, attendance, and other relevant data.

Overall, a well-designed student database enhances efficiency, accuracy, and transparency in managing academic information, supporting informed decision-making, and improving the overall educational experience for students, faculty, and administrators.

Dataset: For this project, I built a sample dataset that contains the following variables:

1. Student_Id
2. Student_Name
3. Age
4. Gender
5. Grade_Level
6. Desk_Id
7. Classroom_Id
8. Classroom_Remark
9. No_Attendnce (A perfect attendance for a semester is 180)
10. Courses_Attendance
11. Percentage
12. Grade_Point_Scale
13. GPA
14. GPA-Remarks
15. Result_Id
16. Student Courses**:**
    (English, Mathematics, Science, Social_Studies, General_Elective, Foreign_Language, Advanced_Placement_(AP)_Courses, Career_And_Technical_Education_(CTE)_Courses).

HILL HIGH SCHOOL DATABASE ERD DIAGRAM FOR MY ANALYSIS



USES OF THE ERD DIAGRM FOR THIS ANALYSIS

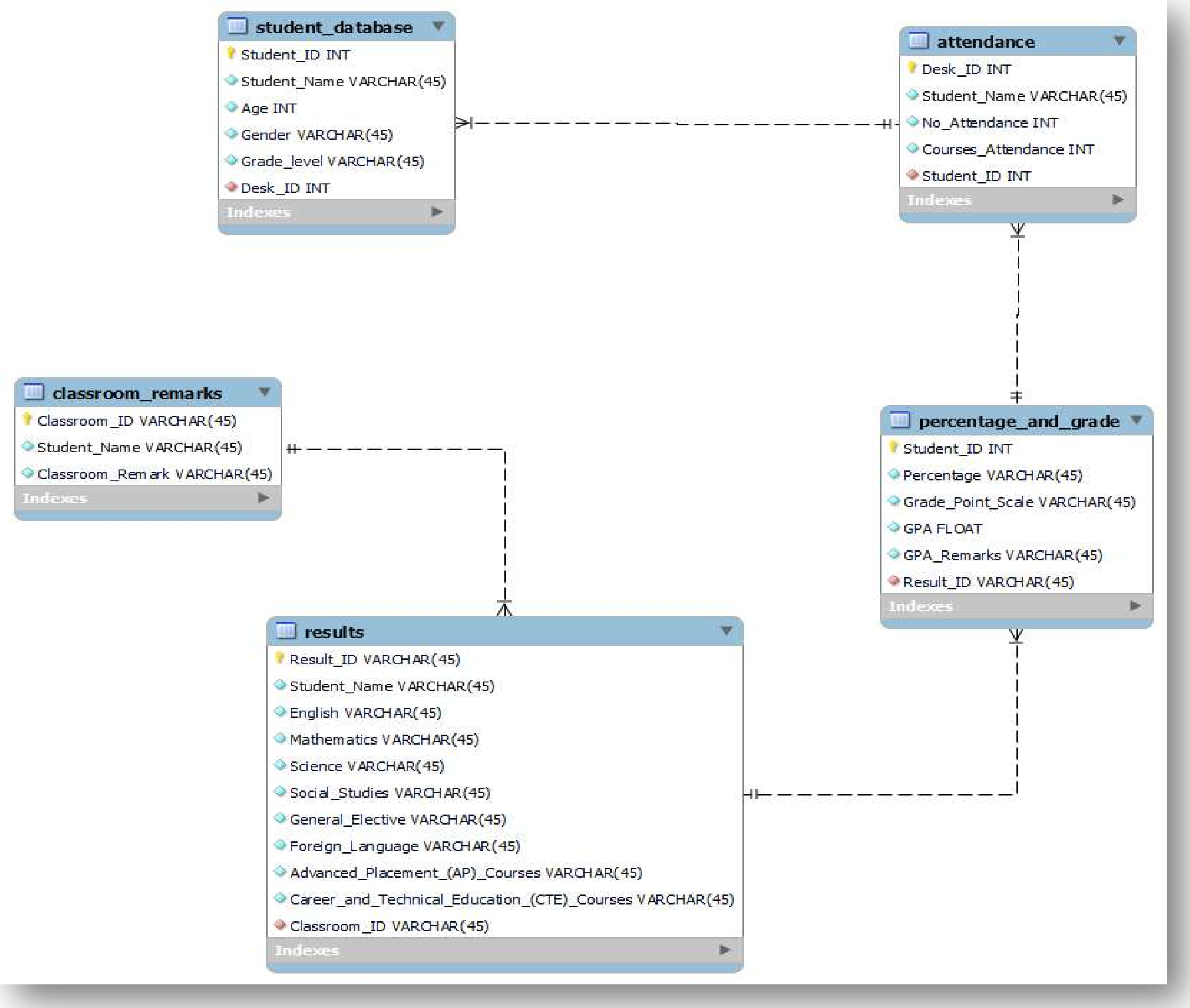
1. To have a full picture of the entire database schema
2. To help in visualizing the entities, their attributes, and the relationships between them.
3. To help in defining the relationships between tables
4. To understand the kind of datatype in each column
5. To provide a clear, visual representation of data structures.

DATABASE FOR MY ANALYSIS: MySQL (Workbench)

WHY USING MYSQL REVERSE ENGINEER FOR GENERATING AN ERD DIAGRAM

The ERD serves as documentation for the MySQL database, capturing its structure and relationships in a clear and concise format. The ERD generated through reverse engineering provides a visual representation of the database schema, making it easier to understand and analyze the relationships between different entities.

HILL HIGH SCHOOL MYSQL ERD DIAGRAM USING REVERSE ENGINEER



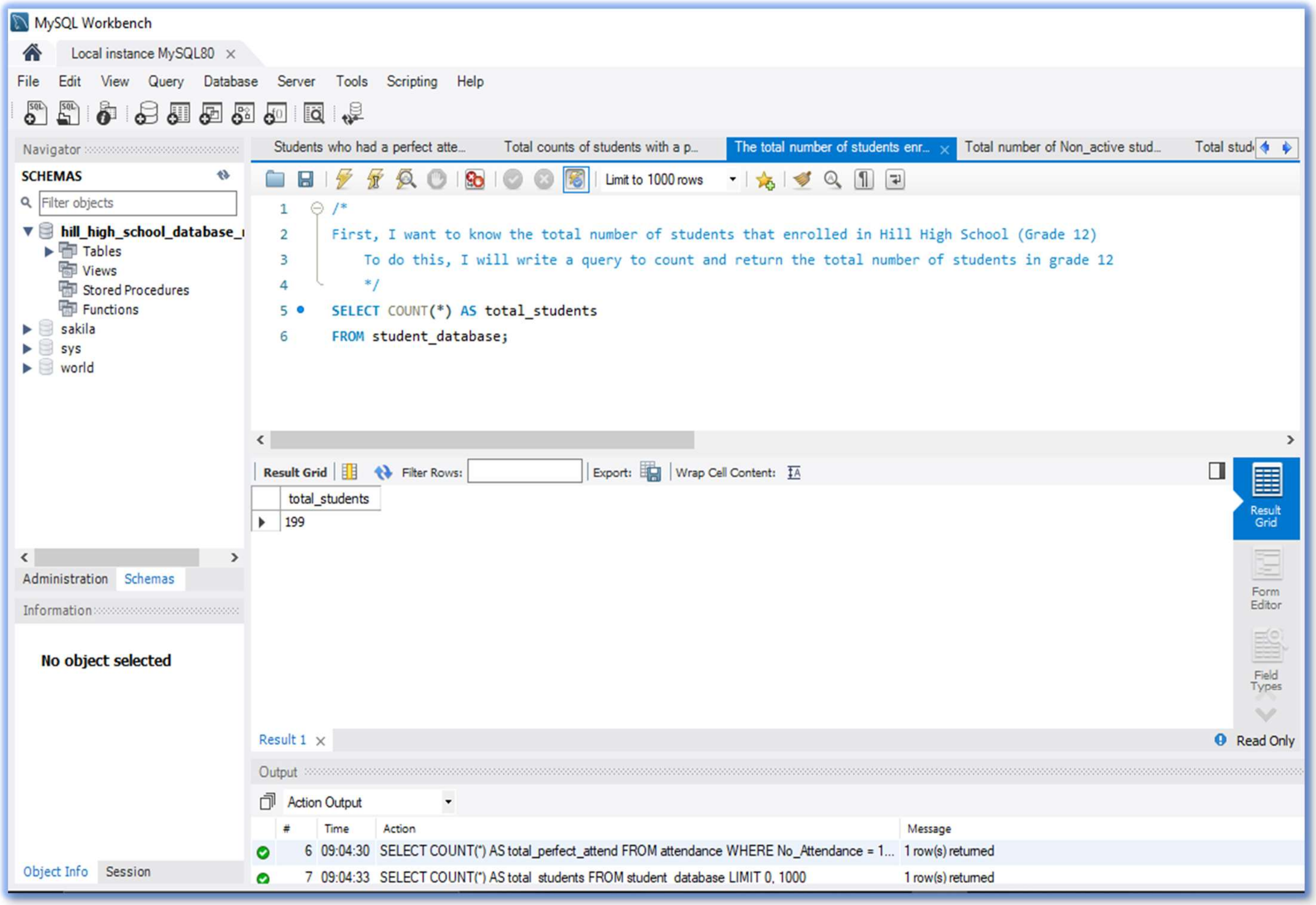
QUESTION(S) TO BE ANSWERED

1. What is the total number of students in Grade 12?
2. What is the total count of students who had a perfect attendance of 180?
3. What is the number of students that had an ‘active’ classroom remark?
4. What is the number of students that had a ‘non-active’ classroom remarks?
5. What is the number of students that had a ‘playful’ classroom remark?
6. Display the student GPAs and their number of attendance and LIMIT by 15
7. Group and count all student GPAs according to their level of performance.
8. How many students had distinctions in their English, Mathematics, and Science courses?
9. How many students failed their English, Mathematics, and Science courses?
10. What is the total number of students in each age group in Grade 12?
11. Students who had a percentage of 97-100 with an A+ as a grade point scale, limit rows to 15.

DATA ANALYSIS

1. What is the Total Number of Students in Grade 12?

Grade 12 is the final class of Hill High School, which after the successful completion of the high school diploma you can then move into college. Using the **SELECT Count (*)** statement we want to know the total number of students who enrolled in Grade 12 at Hill high School.



```
/*
First, I want to know the total number of students that enrolled in Hill High School (Grade 12)
   To do this, I will write a query to count and return the total number of students in grade 12
   */
SELECT COUNT(*) AS total_students
FROM student_database;
```

```
total_students
199
```

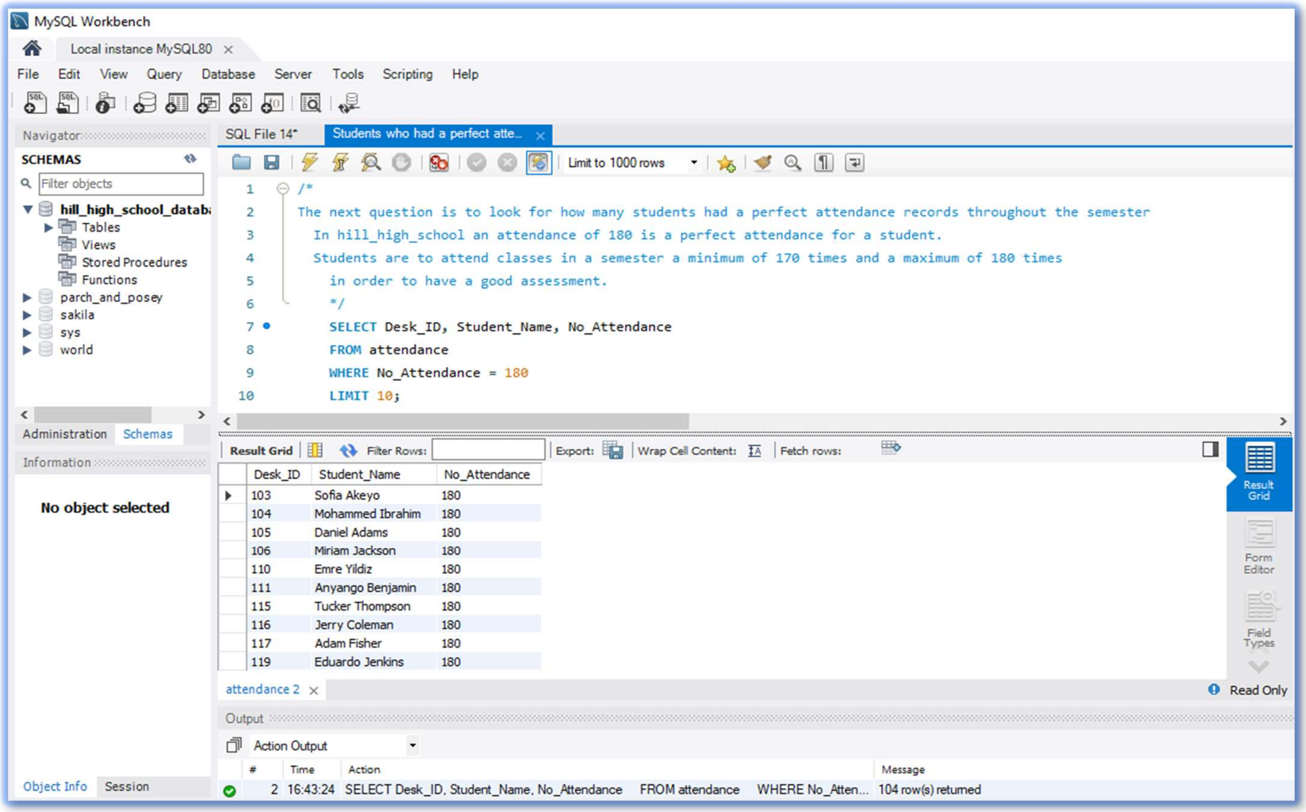
```
#  Time      Action                                                                    Message
6  09:04:30  SELECT COUNT(*) AS total_perfect_attend FROM attendance WHERE No_Attendance = 1...  1 row(s) returned
7  09:04:33  SELECT COUNT(*) AS total_students FROM student_database LIMIT 0, 1000      1 row(s) returned
```

The total number of students in Grade 12 of Hill High School is **199.**

2. What is the Total Count of Students Who had a Perfect Attendance of 180?

The next question is to look for how many students had a perfect attendance record throughout the semester. In Hill High School, an attendance of 180 is a perfect attendance for a student. Students are to attend classes in a semester a minimum of 170 times and a maximum of 180 times to have a good assessment.

Students who had an attendance below 170 may likewise have a poor result and report at the end of the semester. For students to graduate with a high school diploma certificate, there are certain requirements, one of which is having a minimum attendance of 170. Having a maximum of 180 attendance shows the focus and studiousness of that student. So here we are going to find out the students and the total count of students who had perfect attendance of 180.



```sql
/*
The next question is to look for how many students had a perfect attendance records throughout the semester
 In hill_high_school an attendance of 180 is a perfect attendance for a student.
 Students are to attend classes in a semester a minimum of 170 times and a maximum of 180 times
   in order to have a good assessment.
   */
   SELECT Desk_ID, Student_Name, No_Attendance
   FROM attendance
   WHERE No_Attendance = 180
   LIMIT 10;
```

Desk_ID	Student_Name	No_Attendance
103	Sofia Akeyo	180
104	Mohammed Ibrahim	180
105	Daniel Adams	180
106	Miriam Jackson	180
110	Emre Yildiz	180
111	Anyango Benjamin	180
115	Tucker Thompson	180
116	Jerry Coleman	180
117	Adam Fisher	180
119	Eduardo Jenkins	180

```
#   Time      Action                                                                          Message
2   16:43:24  SELECT Desk_ID, Student_Name, No_Attendance  FROM attendance  WHERE No_Atten...  104 row(s) returned
```

In order to find the total count of student who had a perfect attendance of 180, we will have to use the **SELECT COUNT (*)** statement.

We can see the analysis in the diagram below:

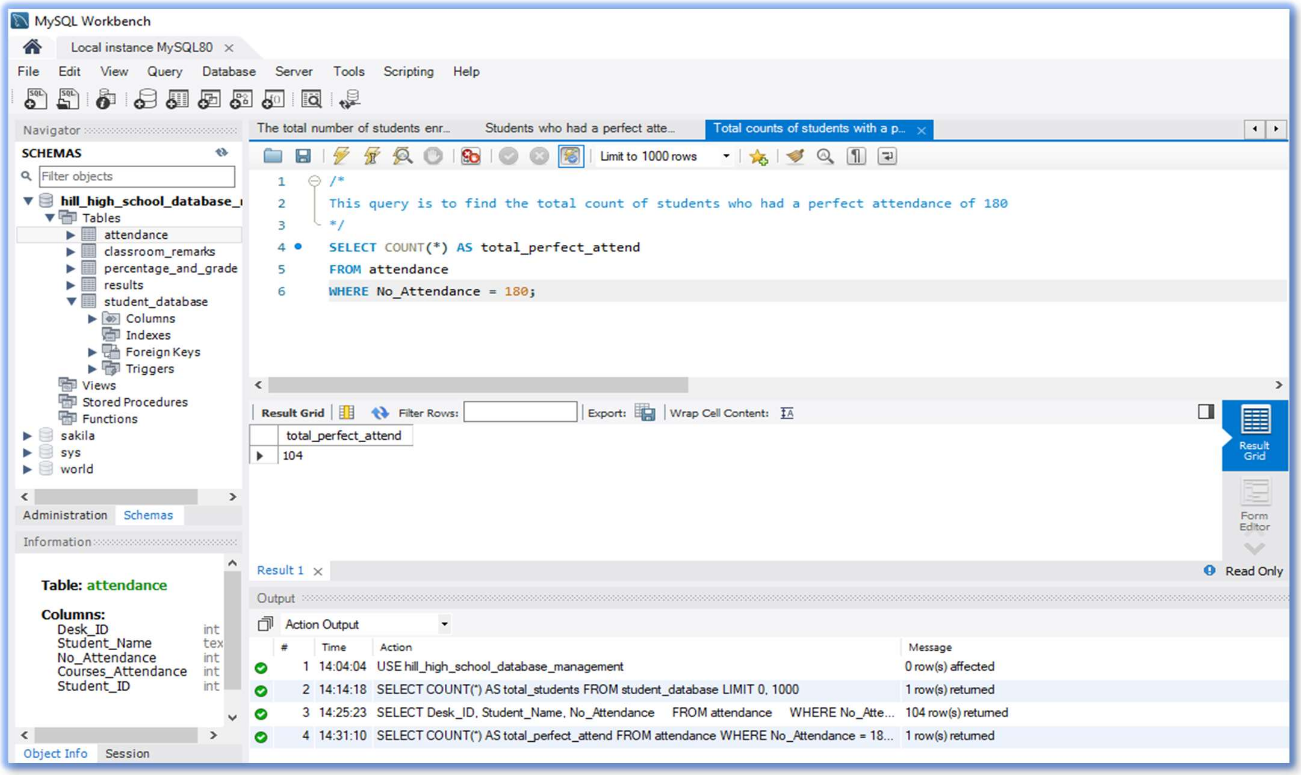


Table: attendance

Columns:

Desk_ID	int
Student_Name	tex
No_Attendance	int
Courses_Attendance	int
Student_ID	int

```sql
/*
This query is to find the total count of students who had a perfect attendance of 180
*/
SELECT COUNT(*) AS total_perfect_attend
FROM attendance
WHERE No_Attendance = 180;
```

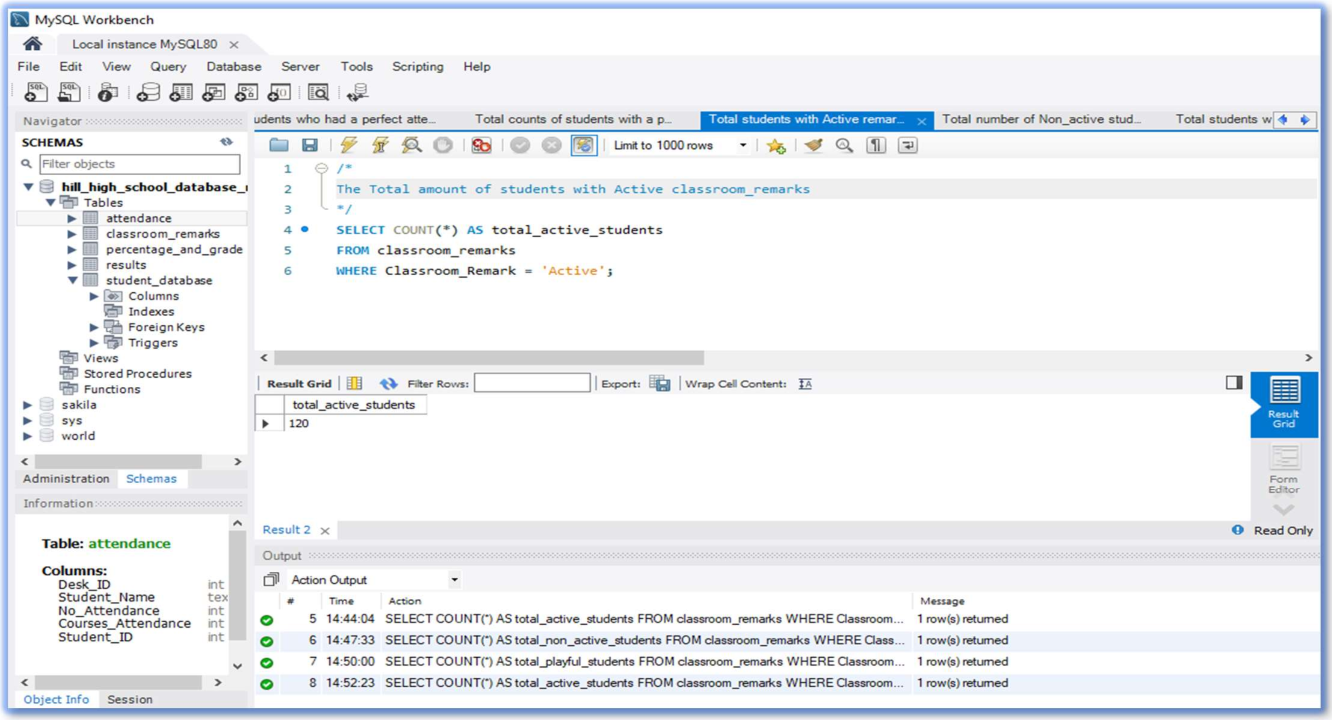
total_perfect_attend
104

```
#   Time      Action                                                                           Message
1   14:04:04  USE hill_high_school_database_management                                         0 row(s) affected
2   14:14:18  SELECT COUNT(*) AS total_students FROM student_database LIMIT 0, 1000            1 row(s) returned
3   14:25:23  SELECT Desk_ID, Student_Name, No_Attendance  FROM attendance  WHERE No_Atte...    104 row(s) returned
4   14:31:10  SELECT COUNT(*) AS total_perfect_attend FROM attendance WHERE No_Attendance = 18... 1 row(s) returned
```

We then noticed that the total amount of students in Grade 12 who had a perfect attendance of 180 was 104 out of a total student of 199. (**104 students had a perfect attendance of 180).**

3. What is the Number of Students that had an ‘Active’ Classroom Remark?

In each grade there must be a classroom remark for each student, here I want to find out the students who had an ‘Active’ classroom remark in Grade 12 which indicates seriousness.



```sql
/*
The Total amount of students with Active classroom_remarks
*/
SELECT COUNT(*) AS total_active_students
FROM classroom_remarks
WHERE Classroom_Remark = 'Active';
```

total_active_students
120

Table: attendance

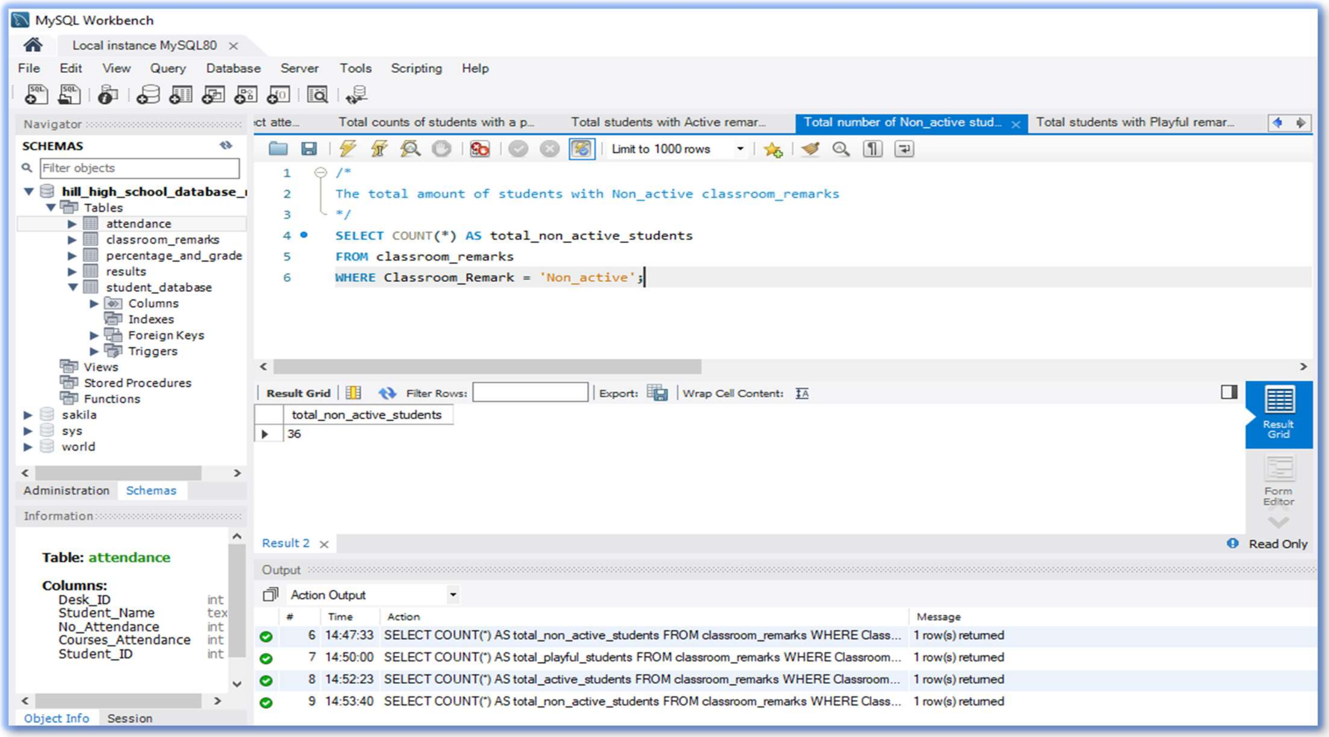
Columns:
Desk_ID int
Student_Name tex
No_Attendance int
Courses_Attendance int
Student_ID int

```
#  Time      Action                                                                                   Message
5  14:44:04  SELECT COUNT(*) AS total_active_students FROM classroom_remarks WHERE Classroom...          1 row(s) returned
6  14:47:33  SELECT COUNT(*) AS total_non_active_students FROM classroom_remarks WHERE Class...        1 row(s) returned
7  14:50:00  SELECT COUNT(*) AS total_playful_students FROM classroom_remarks WHERE Classroom...         1 row(s) returned
8  14:52:23  SELECT COUNT(*) AS total_active_students FROM classroom_remarks WHERE Classroom...          1 row(s) returned
```

The total number of students with an ‘Active’ classroom remark in Hill High School Grade 12 was **120.**

4. What is the Number of Students that had a ‘Non-Active’ Classroom Remark?

I want to find out the students who had a ‘Non-Active’ classroom remark in Grade 12 which indicates unseriousness.



```sql
/*
The total amount of students with Non_active classroom_remarks
*/
SELECT COUNT(*) AS total_non_active_students
FROM classroom_remarks
WHERE Classroom_Remark = 'Non_active';
```

total_non_active_students
36

Table: attendance

Columns:
Desk_ID int
Student_Name tex
No_Attendance int
Courses_Attendance int
Student_ID int

```
#  Time      Action                                                                                   Message
6  14:47:33  SELECT COUNT(*) AS total_non_active_students FROM classroom_remarks WHERE Class...        1 row(s) returned
7  14:50:00  SELECT COUNT(*) AS total_playful_students FROM classroom_remarks WHERE Classroom...         1 row(s) returned
8  14:52:23  SELECT COUNT(*) AS total_active_students FROM classroom_remarks WHERE Classroom...          1 row(s) returned
9  14:53:40  SELECT COUNT(*) AS total_non_active_students FROM classroom_remarks WHERE Class...        1 row(s) returned
```

The total number of students with a ‘Non-Active’ classroom remark in Hill High School Grade 12 was **36.**

5. What is the Number of Students that had a ‘Playful’ Classroom Remark?

Here, I want to find out the total number of students with a ‘Playful’ remark that indicates very unserious behavior.

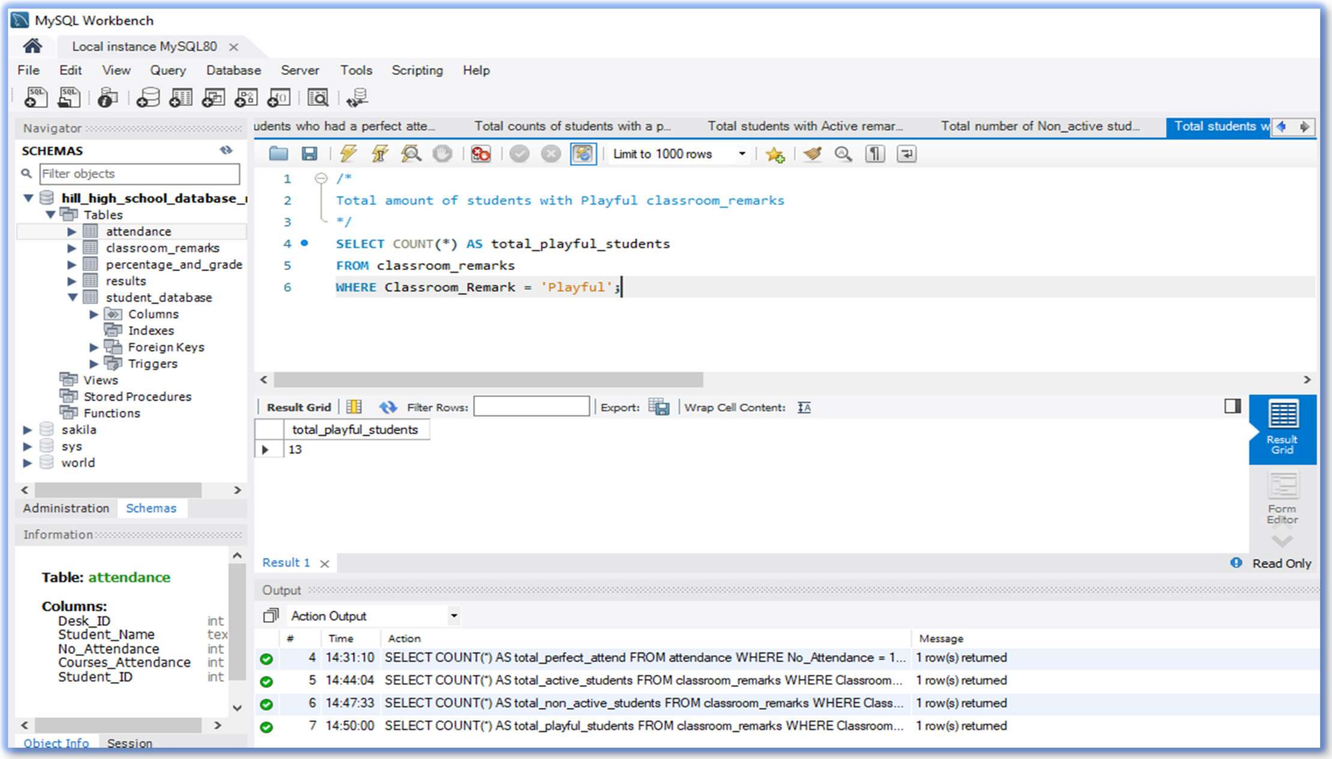


Table: attendance

Columns:

Desk_ID	int
Student_Name	tex
No_Attendance	int
Courses_Attendance	int
Student_ID	int

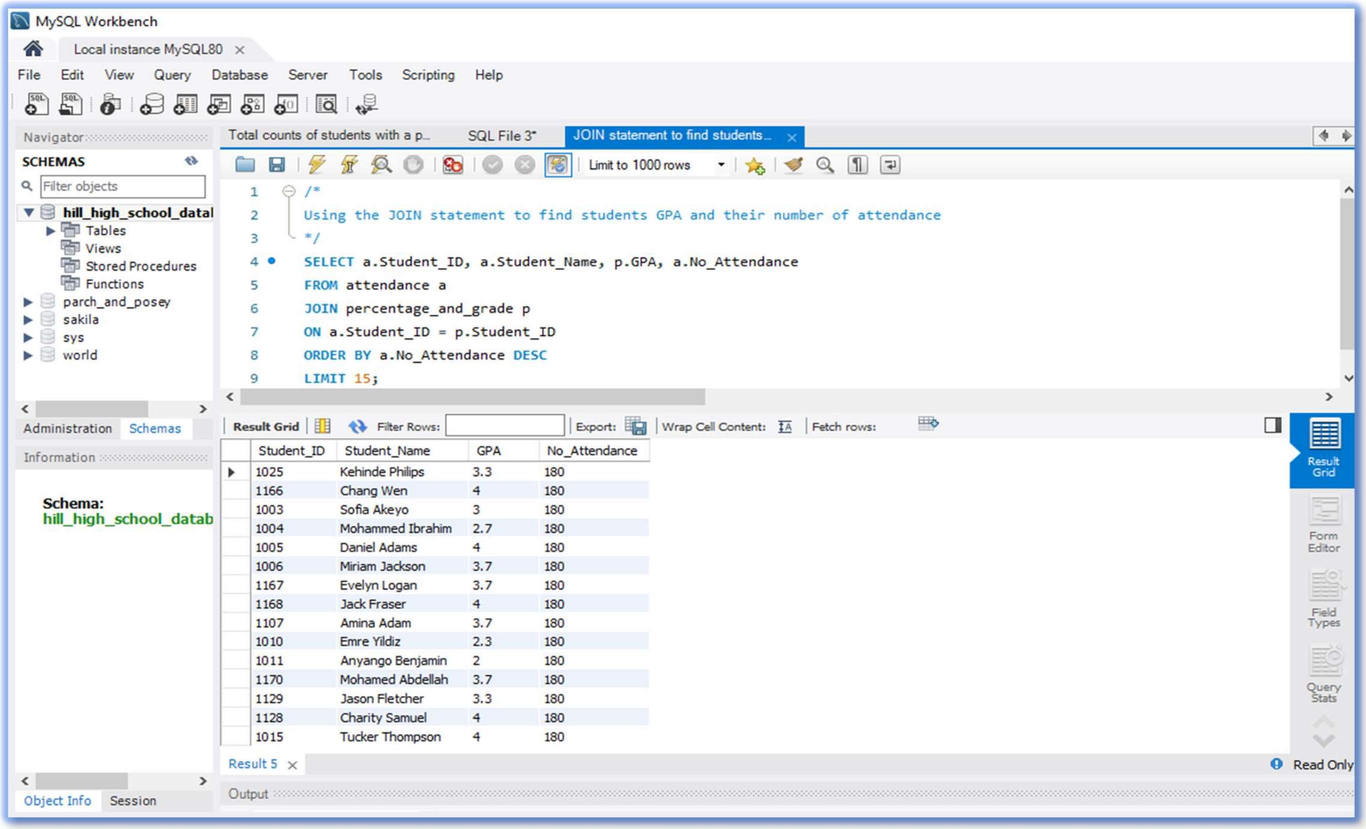
```sql
/*
Total amount of students with Playful classroom_remarks
*/
SELECT COUNT(*) AS total_playful_students
FROM classroom_remarks
WHERE Classroom_Remark = 'Playful';
```

total_playful_students
13

```
 #  Time      Action                                                                              Message
 4  14:31:10  SELECT COUNT(*) AS total_perfect_attend FROM attendance WHERE No_Attendance = 1...  1 row(s) returned
 5  14:44:04  SELECT COUNT(*) AS total_active_students FROM classroom_remarks WHERE Classroom...  1 row(s) returned
 6  14:47:33  SELECT COUNT(*) AS total_non_active_students FROM classroom_remarks WHERE Class...  1 row(s) returned
 7  14:50:00  SELECT COUNT(*) AS total_playful_students FROM classroom_remarks WHERE Classroom...  1 row(s) returned
```

The total number of students that had a ‘Playful’ remark was 13

6. Display the Students GPA and their Number of Attendance, Limit by 15.



Schema:
hill_high_school_datab

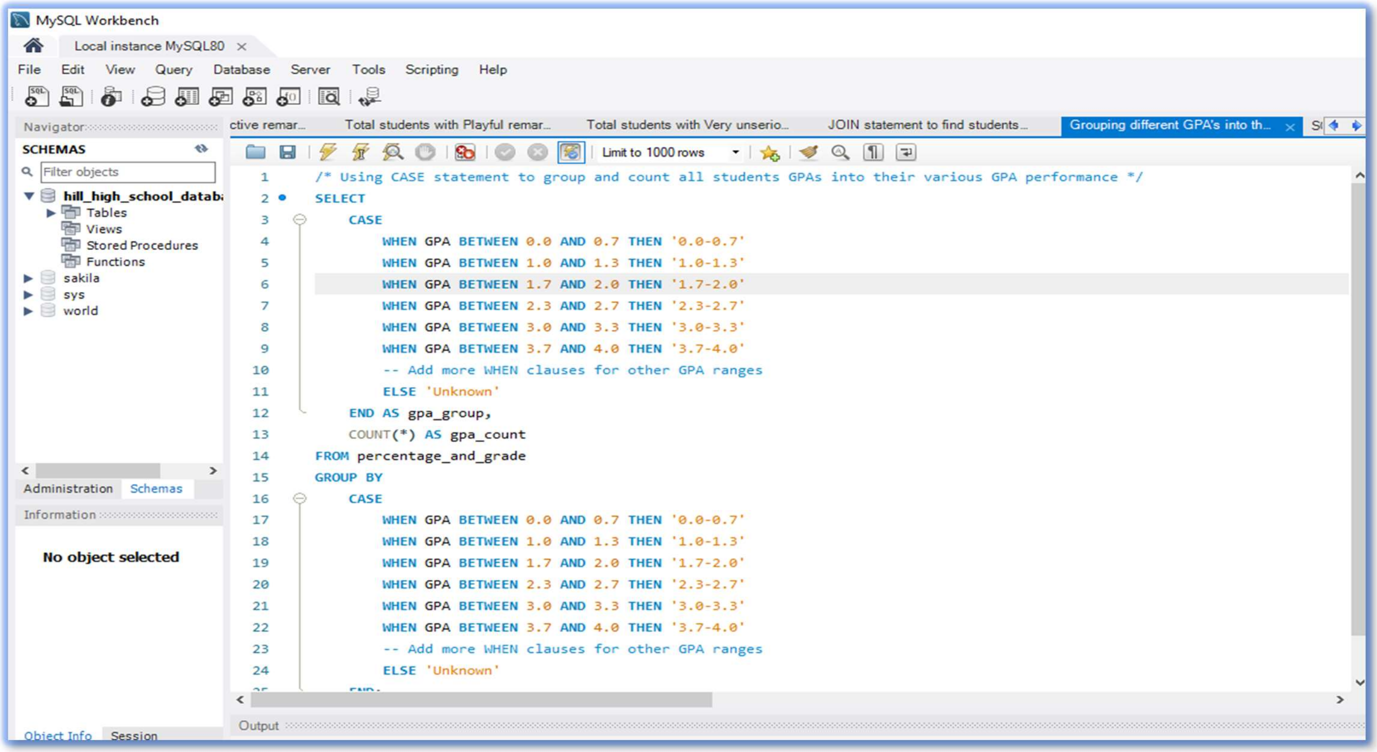
```sql
/*
Using the JOIN statement to find students GPA and their number of attendance
*/
SELECT a.Student_ID, a.Student_Name, p.GPA, a.No_Attendance
FROM attendance a
JOIN percentage_and_grade p
ON a.Student_ID = p.Student_ID
ORDER BY a.No_Attendance DESC
LIMIT 15;
```

Student_ID	Student_Name	GPA	No_Attendance
1025	Kehinde Philips	3.3	180
1166	Chang Wen	4	180
1003	Sofia Akeyo	3	180
1004	Mohammed Ibrahim	2.7	180
1005	Daniel Adams	4	180
1006	Miriam Jackson	3.7	180
1167	Evelyn Logan	3.7	180
1168	Jack Fraser	4	180
1107	Amina Adam	3.7	180
1010	Emre Yildiz	2.3	180
1011	Anyango Benjamin	2	180
1170	Mohamed Abdellah	3.7	180
1129	Jason Fletcher	3.3	180
1128	Charity Samuel	4	180
1015	Tucker Thompson	4	180

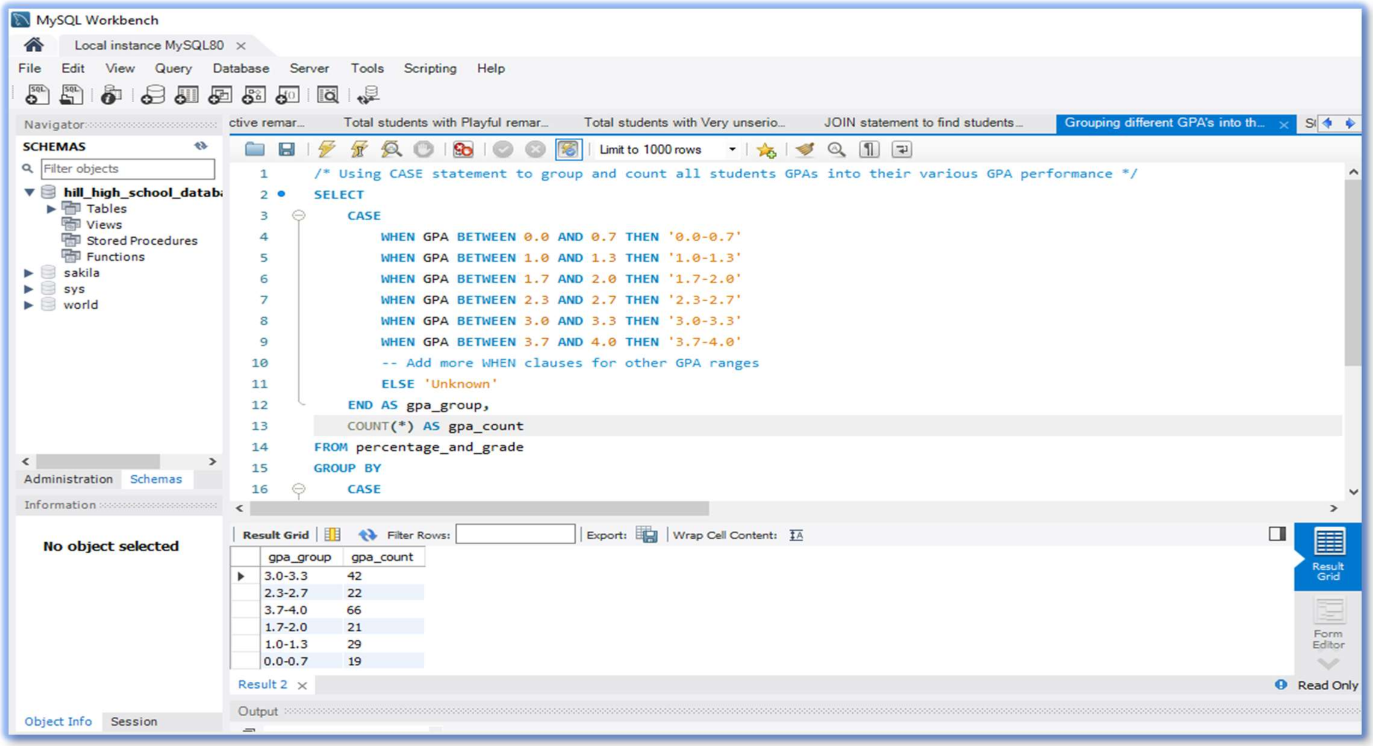
In this section, I used the JOIN statement to join two tables attendance and percentage_and_grade to get the GPA and number of attendance of students, I limited my rows to 15 and sorted my data by the highest number of attendances in that semester. The insights show that not all students with a perfect attendance of 180 had a powerful GPA. This is also vice versa if we sort our data by GPA.

7. Group and Count all Student GPAs According to their Level of Performance.

In this section, I will use a CASE statement to group all the students in grade 12 according to their GPA performance and then count all the students in each category of their grouped performance.



```
/* Using CASE statement to group and count all students GPAs into their various GPA performance */
SELECT
    CASE
        WHEN GPA BETWEEN 0.0 AND 0.7 THEN '0.0-0.7'
        WHEN GPA BETWEEN 1.0 AND 1.3 THEN '1.0-1.3'
        WHEN GPA BETWEEN 1.7 AND 2.0 THEN '1.7-2.0'
        WHEN GPA BETWEEN 2.3 AND 2.7 THEN '2.3-2.7'
        WHEN GPA BETWEEN 3.0 AND 3.3 THEN '3.0-3.3'
        WHEN GPA BETWEEN 3.7 AND 4.0 THEN '3.7-4.0'
        -- Add more WHEN clauses for other GPA ranges
        ELSE 'Unknown'
    END AS gpa_group,
    COUNT(*) AS gpa_count
FROM percentage_and_grade
GROUP BY
    CASE
        WHEN GPA BETWEEN 0.0 AND 0.7 THEN '0.0-0.7'
        WHEN GPA BETWEEN 1.0 AND 1.3 THEN '1.0-1.3'
        WHEN GPA BETWEEN 1.7 AND 2.0 THEN '1.7-2.0'
        WHEN GPA BETWEEN 2.3 AND 2.7 THEN '2.3-2.7'
        WHEN GPA BETWEEN 3.0 AND 3.3 THEN '3.0-3.3'
        WHEN GPA BETWEEN 3.7 AND 4.0 THEN '3.7-4.0'
        -- Add more WHEN clauses for other GPA ranges
        ELSE 'Unknown'
```



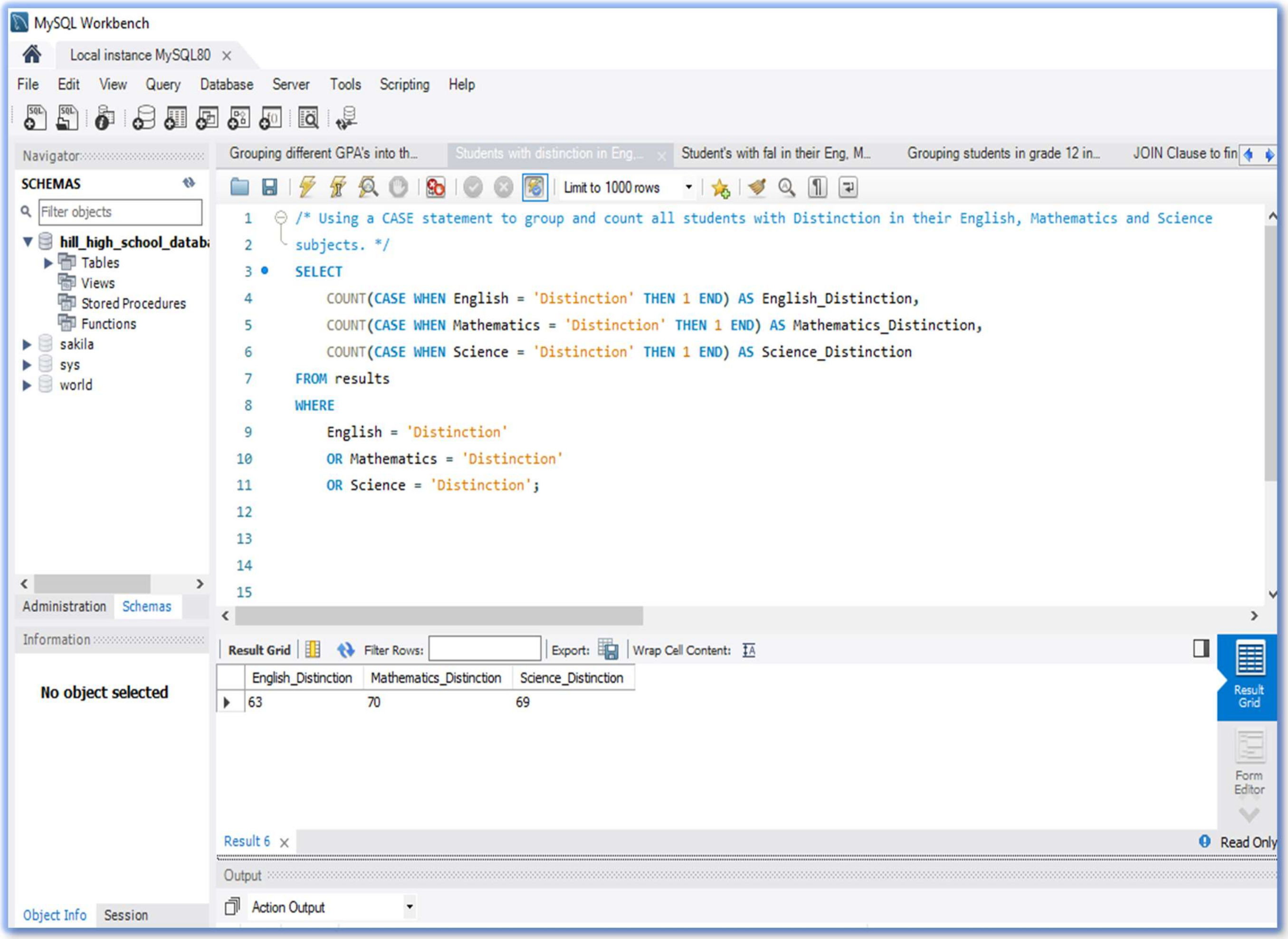
```
/* Using CASE statement to group and count all students GPAs into their various GPA performance */
SELECT
    CASE
        WHEN GPA BETWEEN 0.0 AND 0.7 THEN '0.0-0.7'
        WHEN GPA BETWEEN 1.0 AND 1.3 THEN '1.0-1.3'
        WHEN GPA BETWEEN 1.7 AND 2.0 THEN '1.7-2.0'
        WHEN GPA BETWEEN 2.3 AND 2.7 THEN '2.3-2.7'
        WHEN GPA BETWEEN 3.0 AND 3.3 THEN '3.0-3.3'
        WHEN GPA BETWEEN 3.7 AND 4.0 THEN '3.7-4.0'
        -- Add more WHEN clauses for other GPA ranges
        ELSE 'Unknown'
    END AS gpa_group,
    COUNT(*) AS gpa_count
FROM percentage_and_grade
GROUP BY
    CASE
```

gpa_group	gpa_count
3.0-3.3	42
2.3-2.7	22
3.7-4.0	66
1.7-2.0	21
1.0-1.3	29
0.0-0.7	19

From this insight we can see that students in Grade 12 did extremely well. I noticed this after using the CASE statement to group each student into their various GPA range. The GPA range of 3.7 – 4.0 had the highest number of students, with 66 students having a GPA range between 3.7 – 4.0.

8. How Many Students had Distinctions in their English, Mathematics, and Science Courses?

Here, I would like to know how many students had distinctions in their English, Mathematics, and Sciences courses. Distinction indicates that the student did very well in that course. I will use a CASE statement to get my insights.



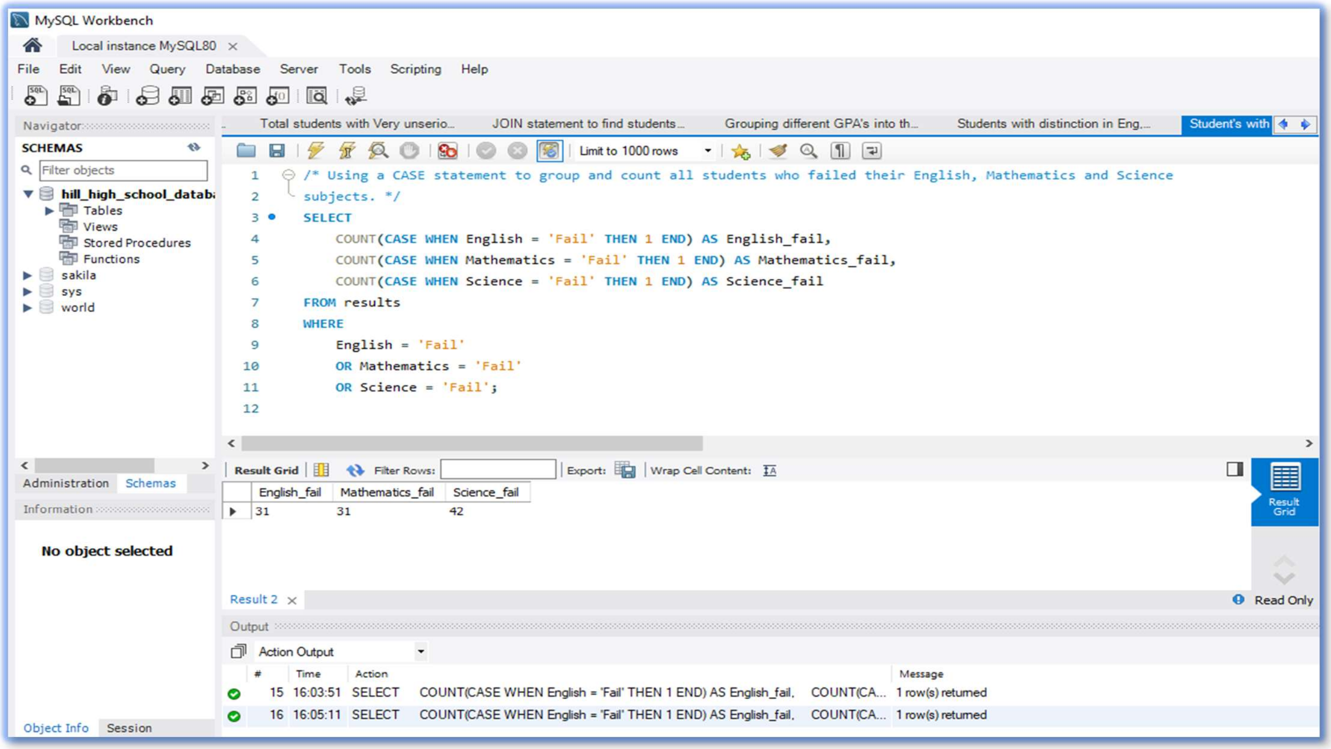
```sql
/* Using a CASE statement to group and count all students with Distinction in their English, Mathematics and Science
subjects. */
SELECT
    COUNT(CASE WHEN English = 'Distinction' THEN 1 END) AS English_Distinction,
    COUNT(CASE WHEN Mathematics = 'Distinction' THEN 1 END) AS Mathematics_Distinction,
    COUNT(CASE WHEN Science = 'Distinction' THEN 1 END) AS Science_Distinction
FROM results
WHERE
    English = 'Distinction'
    OR Mathematics = 'Distinction'
    OR Science = 'Distinction';
```

English_Distinction	Mathematics_Distinction	Science_Distinction
63	70	69

Our analysis showed that out of 199 students in Grade 12**, 63** students had a distinction in English, **70** students had distinctions in Mathematics, and **69** students had distinctions in Science. Mathematics had the best performance among these three subjects in Grade 12.

9. How Many Students Failed their English, Mathematics, and Science Courses?

This query statement is to find out the total number of students in Grade 12 who failed their English, Mathematics and Science courses



```sql
/* Using a CASE statement to group and count all students who failed their English, Mathematics and Science
subjects. */
SELECT
    COUNT(CASE WHEN English = 'Fail' THEN 1 END) AS English_fail,
    COUNT(CASE WHEN Mathematics = 'Fail' THEN 1 END) AS Mathematics_fail,
    COUNT(CASE WHEN Science = 'Fail' THEN 1 END) AS Science_fail
FROM results
WHERE
    English = 'Fail'
    OR Mathematics = 'Fail'
    OR Science = 'Fail';
```

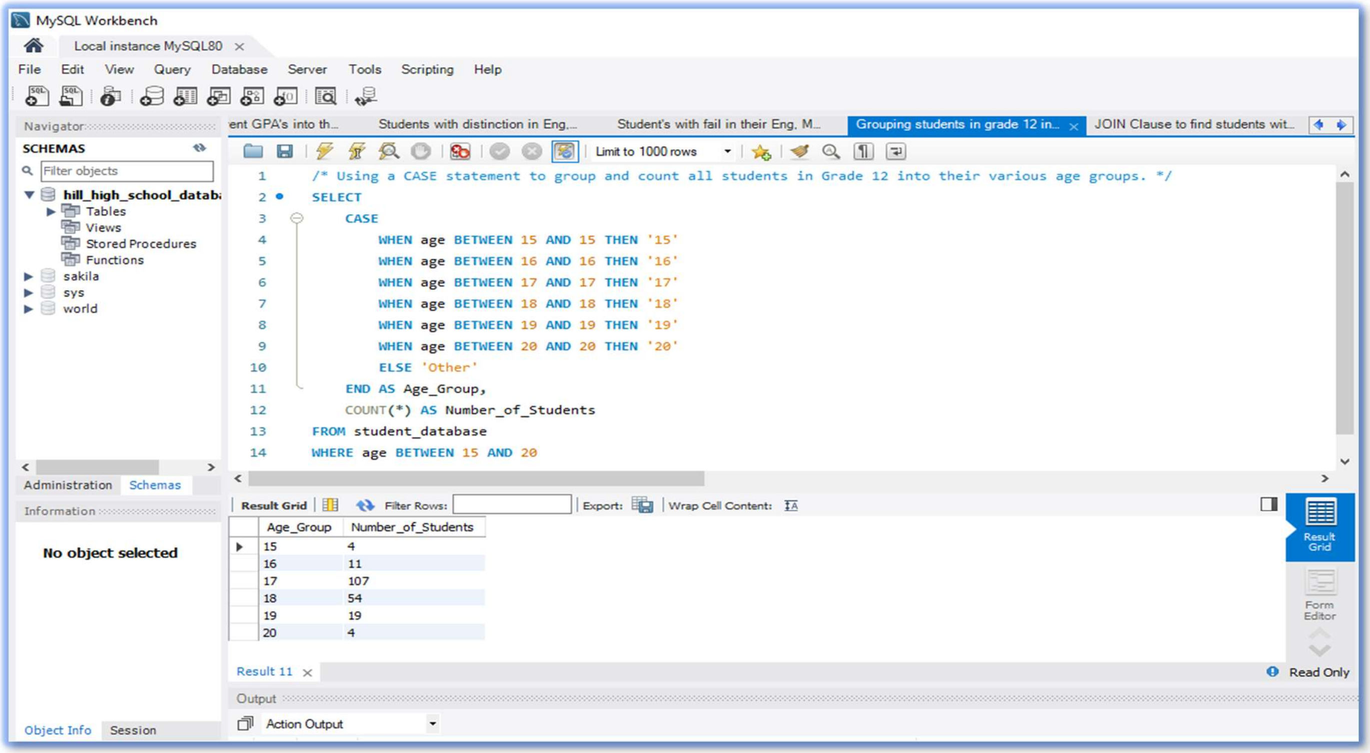
English_fail	Mathematics_fail	Science_fail
31	31	42

```
#   Time      Action  Message
15  16:03:51  SELECT  COUNT(CASE WHEN English = 'Fail' THEN 1 END) AS English_fail,  COUNT(CA...  1 row(s) returned
16  16:05:11  SELECT  COUNT(CASE WHEN English = 'Fail' THEN 1 END) AS English_fail,  COUNT(CA...  1 row(s) returned
```

Our analysis showed that out of 199 students in Grade 12, **31** students failed English, **31** students had failed Mathematics, and 42 students failed Science. Science had the lowest performance among these three subjects in Grade 12.

10. What is the Total number of Students in each Age Group in Grade 12?

In this section, I want to know the total number of students in each age group, I will be using a CASE statement to group each student into their age group.

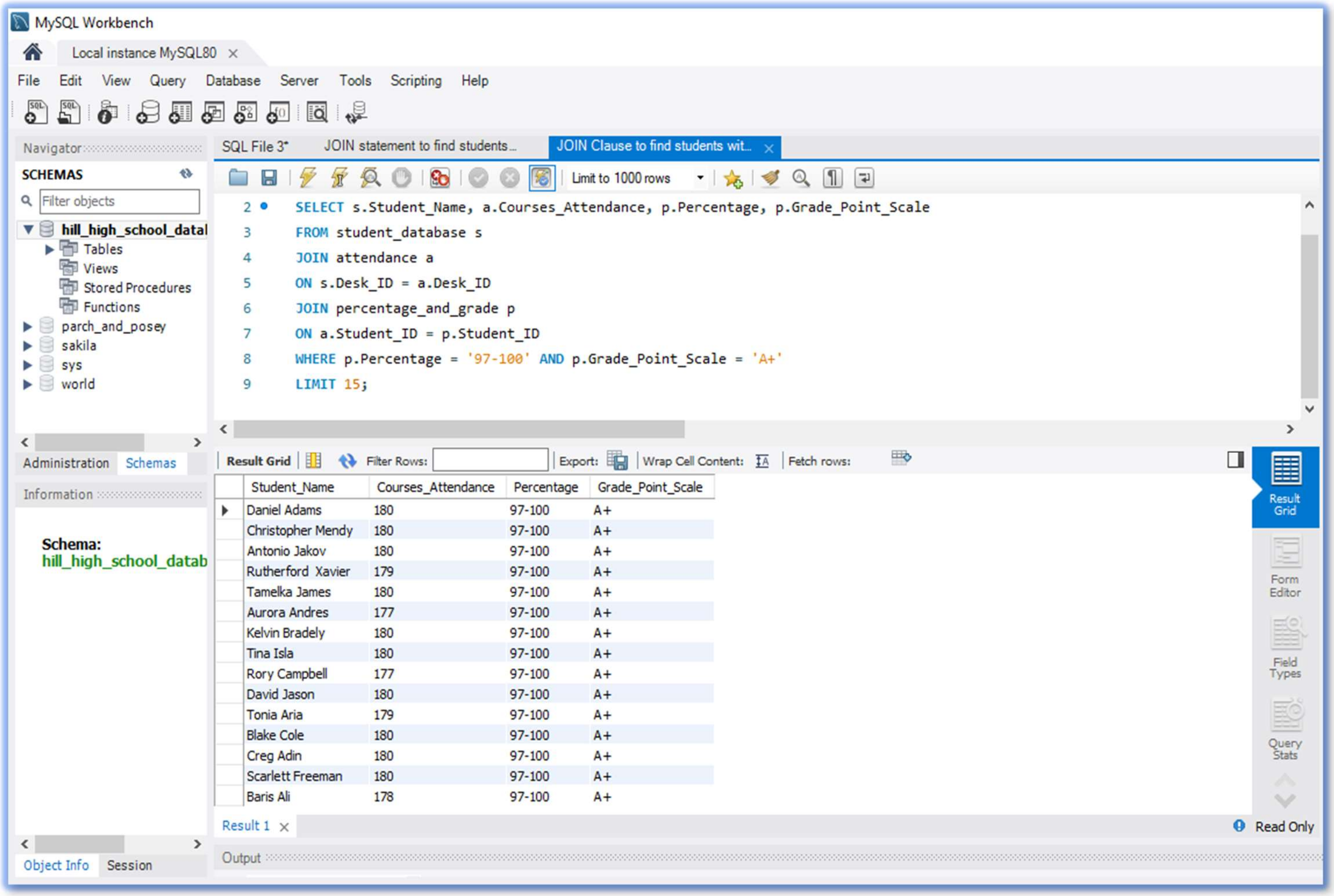


```sql
/* Using a CASE statement to group and count all students in Grade 12 into their various age groups. */
SELECT
    CASE
        WHEN age BETWEEN 15 AND 15 THEN '15'
        WHEN age BETWEEN 16 AND 16 THEN '16'
        WHEN age BETWEEN 17 AND 17 THEN '17'
        WHEN age BETWEEN 18 AND 18 THEN '18'
        WHEN age BETWEEN 19 AND 19 THEN '19'
        WHEN age BETWEEN 20 AND 20 THEN '20'
        ELSE 'Other'
    END AS Age_Group,
    COUNT(*) AS Number_of_Students
FROM student_database
WHERE age BETWEEN 15 AND 20
```

Age_Group	Number_of_Students
15	4
16	11
17	107
18	54
19	19
20	4

In this analysis, I found out that the age group of 17 had the highest number of students. That tells us that age 17 had the highest number of students in the Grade 12 final class. While age 18 had the second highest number of students in Grade 12 final class.

11. Students Who had a Percentage of 97-100 with an A+ as a Grade Point Scale, Limit Rows to 15.



```
SELECT s.Student_Name, a.Courses_Attendance, p.Percentage, p.Grade_Point_Scale
FROM student_database s
JOIN attendance a
ON s.Desk_ID = a.Desk_ID
JOIN percentage_and_grade p
ON a.Student_ID = p.Student_ID
WHERE p.Percentage = '97-100' AND p.Grade_Point_Scale = 'A+'
LIMIT 15;
```

Student_Name	Courses_Attendance	Percentage	Grade_Point_Scale
Daniel Adams	180	97-100	A+
Christopher Mendy	180	97-100	A+
Antonio Jakov	180	97-100	A+
Rutherford Xavier	179	97-100	A+
Tamelka James	180	97-100	A+
Aurora Andres	177	97-100	A+
Kelvin Bradely	180	97-100	A+
Tina Isla	180	97-100	A+
Rory Campbell	177	97-100	A+
David Jason	180	97-100	A+
Tonia Aria	179	97-100	A+
Blake Cole	180	97-100	A+
Creg Adin	180	97-100	A+
Scarlett Freeman	180	97-100	A+
Baris Ali	178	97-100	A+

These are the names of the students in Grade 12 who had a percentage of 97-100 and a grade point scale of A+. From this analysis, you will notice students who had good course attendance for the semester had a grade point scale of A+, even if we don't limit our rows to 15 that will still be our observation.

Number of School Attendance = Number of Course Attendance

- Maximum Number of School Attendance = 180
- Maximum Number of Course Attendance = 180
- Minimum Number of School Attendance = 170
- Minimum Number of Course Attendance = 170

GENERAL INSIGHTS FROM HILL HIGH SCHOOL (GRADE 12) DATABASE

1. 199 students enrolled at Hill High School Grade 12
2. 104 students in Hill High School Grade 12 had a perfect attendance of 180
3. 120 students were very active in the class
4. 36 students were non-active in the class
5. 13 students were very playful in the class
6. Many students had a good performance based on their GPA
7. The GPA range of 3.7 – 4.0 had the highest number of students, with 66 students having a GPA range between 3.7 – 4.0, making the class academic performance a very good one
8. Mathematics course had more students with distinctions (based on the three most important courses, English, Mathematics, and Science).
9. Science course had more students who failed (based on the three most important courses, English, Mathematics, and Science).
10. The age group of 17 years had the highest number of enrolled students in Grade 12.
11. Most students who had a minimum course attendance of 170 and maximum course attendance of 180 had a percentage of 97-100 and a Grade Point Scale of A+.